IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION FOR PATENT

MEMORY UNIT WITH CONTROLLER MANAGING MEMORY ACCESS THROUGH JTAG AND CPU INTERFACES

Inventors: Mitrajit Chatterjee,
Ming Tang,
Peter Z. Onufryk, and
Steven Chau.

FIELD OF THE INVENTION

[0001] The present invention generally relates to memory controllers and in particular, to a memory unit with controller managing memory access through JTAG and CPU interfaces.

BACKGROUND OF THE INVENTION

[0002] In certain applications it is useful to restrict access to memory such as when the memory stores secret information like a private key for decryption purposes or a digital signature for authorization purposes. Although access to read such information may be restricted, it may be useful to allow multiple sources to write such information into the memory. Also, once the information is written, it is useful for security or other purposes to prevent the information from being maliciously or inadvertently overwritten.

[0003] Non-volatile memories are useful in these and other applications, because they maintain stored information even after power is turned off. Electrically programmable, non-volatile memories have the additional

advantage of being programmable after their manufacture. This allows for easy customization at the system level.

[0004] A central processing unit ("CPU") is commonly used to program an electrically programmable, non-volatile memory. However, when information to be stored in the memory is necessary for the proper operation of the CPU, or is to be programmed before the CPU is booted-up or otherwise programmed without the CPU running at the time, another means for programming the memory in such cases is needed.

OBJECTS AND SUMMARY OF THE INVENTION

[0005] With the Joint Test Action Group ("JTAG") debugging techniques, electronic devices gather information about their own operation and route the information to external pins so that the information may be accessed by external JTAG hardware.

[0006] Various aspects of the present invention take advantage of the observation that the external JTAG hardware might also be used as an alternative to the CPU as a means to configure the memory and/or program it with information.

[0007] Accordingly, it is an object of the present invention to provide a memory unit with controller managing access to memory through JTAG and CPU interfaces.

[0008] Another object is to provide a memory unit wherein writing to a protected area of the memory is freely

allowed and reading the written information from the protected area is restricted.

[0009] Another object is to provide a memory unit with controller that is capable of storing information in memory prior to the memory unit or its system leaving an assembly line.

[0010] Another object is to provide a memory unit with controller that is capable of storing information in memory prior to a CPU coupled to the memory unit booting up.

[0011] These and other objects are accomplished by the various aspects of the present invention, wherein briefly stated, one aspect is a memory unit comprising a memory and a controller, wherein the controller is configured to provide unrestricted access for writing information into an unwritten area of the memory and restricted access for reading the written information.

[0012] Another aspect is a memory unit comprising a memory including a protected area, a JTAG interface, logic, and a controller. The controller is configured to allow JTAG hardware to write information into the protected area through the JTAG interface, and allow the logic exclusive access to read the written information.

[0013] Another aspect is a memory unit comprising a memory, a JTAG interface, and a controller. The JTAG interface is clocked by a JTAG clock signal received from a external JTAG hardware. The controller is configured to allow the external JTAG hardware to write information into the memory through the JTAG interface. The controller is

clocked by a system clock signal if the system clock signal is available or by the JTAG clock signal if the system clock signal is not available.

[0014] Yet another aspect is a memory unit comprising a non-volatile memory storing a first boot configuration vector, and a reset circuit coupled to a plurality of external boot configuration vector pins. When the boot configuration vector is to be provided internally (i.e., from within an integrated circuit device including the memory unit), a designated one of the external boot configuration vector pins is not asserted (e.g., not tied to a HIGH logic state). When the boot configuration vector is to be provided externally (i.e., from outside the integrated circuit device), the designated one of the external boot configuration vector pins is asserted (e.g., tied to a HIGH logic state) and a second boot configuration vector is provided on others of the external boot configuration vector pins. Consequently, the reset circuit first checks the status of the designated one of the external boot configuration vector pins upon its activation. If the status is in a first state (e.g., not asserted), the reset circuit generates one or more initialization signals using the first boot configuration vector. On the other hand, if the status is in a second state (e.g., asserted), the reset circuit generates its one or more initialization signals using the second boot configuration vector.

[0015] Additional objects, features and advantages of the various aspects of the present invention will become apparent from the following description of its preferred

embodiment, which description should be taken in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0016] **FIG. 1** illustrates a block diagram of a microprocessor system utilizing aspects of the present invention.

[0017] **FIG. 2** illustrates a memory organization for a non-volatile memory utilizing aspects of the present invention.

[0018] **FIG. 3** illustrates a flow diagram of a method for selecting a clock signal in a memory unit utilizing aspects of the present invention.

[0019] **FIG. 4** illustrates a flow diagram of a method for providing a boot configuration vector utilizing aspects of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

[0020] **FIG. 1** illustrates, as an example, a block diagram of a portion of a Microprocessor System including various components on and off a Chip **100**. An on-chip Memory Unit **101** includes a Non-Volatile Random-Access-Memory ("NVRAM") **102**, a JTAG Interface ("JTAG I/F") **103**, a CPU Interface ("CPU I/F") **104**, a Controller **105**, and Authorization Logic **106**. Other on-chip components include a CPU **107** coupled to the Controller **105** through the CPU I/F **104**, a Clock Generator ("CLK GEN") **110** which generates a System Clock Signal ("SYS_CLK") used for clocking the CPU **107** and other components on the Chip **100**, and a Reset Circuit **114**.

[0021] Off-chip components include System Memory **109** which is conventionally coupled to the CPU **107** through a System Bus **108**, External JTAG Pins **121** which are coupled to the JTAG I/F **103**, and External Boot Configuration Vector ("BCV") Pins **115** which are coupled to the Reset Circuit **114**.

[0022] External JTAG Hardware ("JTAG H/W") **122** is coupled at times (as indicated by a dotted-arrow connection) to the JTAG I/F **103** through the External JTAG Pins **121** for debugging the Chip **100** and other purposes. The hardware is referred to as being JTAG H/W, because it follows the JTAG protocol in communicating with the JTAG I/F **103**. On the other hand, a Dip Switch **123** (or other external hardware device) may be coupled (as indicated by a dotted-arrow connection) to the External BCV Pins **115** and manually or otherwise set for providing a BCV to the Reset Circuit **114** upon its request when a designated one of the External BCV Pins **115** is asserted by the Dip Switch **123**. When the designated one of the External BCV Pins **115** is not asserted by the Dip Switch **123**, however, a BCV stored in the NVRAM **102** is provided to the Reset Circuit **114** upon its request.

[0023] In normal operation, the Controller **105** is clocked by the System Clock Signal generated by the Clock Generator **110**. Since the JTAG I/F **103** is clocked with a JTAG clock signal provided by the JTAG H/W **122**, a First Input Buffer **111** is used in a conventional fashion to synchronize the signals from the JTAG I/F **103** with signals in the Controller **105** (i.e., clocking them into the First Input Buffer **111** using the JTAG Clock Signal and clocking

them out of the First Input Buffer **111** using the System Clock Signal), because the two clock signals may be unrelated to each other. A First Output Buffer (not shown) is also included in the Controller **105** for synchronizing signals going in the opposite direction between the Controller **105** and the JTAG I/F **103**.

[0024] In a preferred embodiment, an IP Bus **130** is clocked with an IP-Bus Clock Signal which is also internally generated on the Chip **100**. In this case, a Second Input Buffer **112** is used in a conventional fashion to synchronize the signals from the CPU I/F **104** with signals in the Controller **105**. A Second Output Buffer (not shown) is also included in the Controller **105** for synchronizing signals going in the opposite direction between the Controller **105** and the CPU I/F **104**. A corresponding pair of buffers is also included in the CPU I/F **104** for synchronizing signals between the IP Bus **130**, and the System Bus **109** which is clocked by the System Clock Signal.

[0025] An example of a memory organization for the NVRAM **102** is shown in **FIG. 2**. The NVRAM **102** in this case is preferably a flash EEPROM including both non-protected and protected areas. The non-protected areas include areas for storing upper and lower bytes of a Boot Configuration Vector ("BCV"), respectively **202** and **203**, and areas **204** and **206** for general or otherwise non-reserved purposes.

[0026] The Controller **105** freely allows read and write access to the non-protected areas **202**, **203**, **204** and **206**. Therefore, the CPU **107** is free to write information to and read information from these non-protected areas through the

CPU Interface **104**, and the JTAG H/W **122** and other JTAG related components such as a Scan Path are also free to write to and/or read from these non-protected areas through the JTAG Interface **103**.

[0027] Although the CPU **107** may write to the upper and lower bytes of the BCV **202** and **203** at any time, the Memory Unit **101** can be advantageously configured so as to allow the external JTAG H/W **122** to write to these two bytes prior to boot-up of the CPU **107**. Since the System Clock signal is not available prior to boot-up of the CPU **107**, the Controller **105** uses the JTAG Clock Signal provided by the external JTAG H/W **122** for its operation in this case.

[0028] A simple method for determining which clock signal to use is illustrated in **FIG. 3**. In **301** of that figure, the Controller **105** receives a start up indication through the JTAG I/F **103** or other source on the Chip **100**. In **302**, it (or another component on the Chip **100**) determines whether the System Clock Signal is available. If the answer is YES, then in **303**, it uses the System Clock Signal for its operation. On the other hand, if the answer is NO, then in **304**, it uses the JTAG Clock Signal.

[0029] On reset, it may be desirable for the Reset Circuit **114** to use a different BCV than the one stored in the NVRAM **102** to generate its one or more initialization (i.e., initial configuration) signals without over-writing the BCV stored in the NVRAM **102**. The Reset Circuit **114** accommodates this by being configured to perform a method illustrated in **FIG. 4**. In **401** of that figure, the Reset Circuit **114** is activated by receiving a reset indication. In **402**, it checks whether a designated one of the External

BCV Pins **115** is asserted (e.g., by an external device pulling it to a logic HIGH state). In **403**, a decision is made based upon whether the designated one of the External BCV Pins **115** is asserted. If the answer is NO, then in **404**, the Reset Circuit **114** uses the BCV stored in the NVRAM **102** to generate its initialization signals. On the other hand, if the answer is YES, then in **405**, the Reset Circuit **114** uses a BCV provided on other of the External BCV Pins **115** by the JTAG H/W **122** or other component coupled to the External BCV Pins **115** at the time to generate its initialization signals.

[0030] Reset is handled in the Memory Unit **101** in the following manner. The JTAG I/F **103** is reset on a System-Reset, Power-On-Reset, or when a JTAG-Reset Pin is asserted (which is one of the External Pins **121**). The Controller **105**, the Authorization Logic **106**, and the NVRAM **101**, on the other hand, reset on a System-Reset, Power-On-Reset, or an Internal Reset generated by the JTAG I/F **103**. The JTAG I/F **103** generates the Internal Reset if the JTAG-Reset Pin is asserted and the System Clock Signal is not present. The JTAG I/F **103** does not generate the Internal Reset if the System Clock Signal is present. This allows the Controller **105**, the Authorization Logic **106**, and the NVRAM **101** to function normally when there is a System Clock Signal (such as the CPU **107** accessing the Controller **105**) when the JTAG I/F **103** is being reset by the JTAG-Reset Pin being asserted.

[0031] Referring back to **FIG. 2**, protected areas in the NVRAM **102** include areas for storing an Authorization Unit Information Block ("AUIB") **205** and an Authorization Unit

Information Block Pointer ("AUIBPTR") **201**. These protected areas are used in conjunction with the Authorization Logic **106**.

[0032] The function of the Authorization Logic **106** is to ensure that authorized software is locked to (i.e., only runs on) the CPU **107**. Secret information used for such authorization is stored in the AUIB **205**, whose location is specified by writing its initial address into the AUIBPTR **201**.

[0033] The Controller **105** allows either the CPU **107** or the JTAG H/W to write to the AUIBPTR **201** and the AUIB **205**. Once information is written into the AUIBPTR **201** and the AUIB **205**, however, the Controller **105** only allows the Authorization Logic **106** to read their contents, and prevents any over-writing of the information until the NVRAM **102** is entirely erased (e.g., by setting all bits of the NVRAM **102** to zero).

[0034] Although the CPU **107** may write to the AUIBPTR **201** and the AUIB **205** at any time, the Memory Unit **101** can be advantageously configured so as to allow the external JTAG H/W **122** to write to these two areas of the NVRAM **102** while the Chip **100** or a system including the Chip **100** is still in its assembly line (such as during electrical and/or functional testing) and/or prior to boot-up of the CPU **107**. Since the System Clock signal is not available prior to boot-up of the CPU **107**, the Controller **105** uses the JTAG Clock Signal provided by the external JTAG H/W **122** (or other component coupled to the JTAG I/F **103** at the time) for its operation such as previously described in reference to **FIG. 3**.

[0035] Once information is written into the AUIBPTR **201** and the AUIB **205**, any attempt by the CPU **107** or JTAG H/W **122** to read these protected areas of the NVRAM **102** results in receiving all "0's" back from the Controller **105**, regardless of the values stored in the AUIBPTR **201** and the AUIB **205**. Also, if either the CPU **107** or JTAG H/W **122** attempts to over-write these areas, such write operations will be ignored by the Controller **105**.

[0036] For the JTAG I/F **103** to access the non-protected areas of the NVRAM **102** (or perform a first write to the protected areas) after the CPU **107** has been booted up, a JTAG START command is sent by the JTAG H/W **122** or other component coupled to the JTAG I/F **103** at the time. After the Controller **105** sees the JTAG START command, it performs the following actions: (i) it blocks any new access requests received from the CPU I/F **104**, and (ii) waits for a period of a few clock cycles (corresponding to a clock synchronization delay for the Second Input Buffer **112**) after any current access request from the CPU I/F **104** finishes before granting access to the JTAG I/F **103**. The reason that the Controller **105** waits for the few clock cycles before granting access to the JTAG I/F **103** is to accommodate CPU access requests already residing in the Second Input Buffer **112** at the time the Controller **105** sees the JTAG START command.

[0037] If another access request from the CPU I/F **104** starts being processed before the period of the few clock cycles passes, then the Controller **105** loops back through (i) and (ii) above until no such access requests appear from the CPU I/F **104** before the period passes. After the

JTAG H/W **122** or other component coupled to the JTAG I/F **103** is granted access to the NVRAM **102** and completes its operation, it sends a JTAG END command to the Controller **105** through the JTAG I/F **103**. After seeing the JTAG END command, the Controller **105** is once again free to grant any permissible access requests from the CPU I/F **104** for the NVRAM **102**.

[0038] Although the various aspects of the present invention have been described with respect to a preferred embodiment, it will be understood that the invention is entitled to full protection within the full scope of the appended claims.